



National Textile University

Department of Computer Science

Subject:

Operating System

Submitted to:

Sir Nasir Mehmood

Submitted by:

Eman Babar

Reg. number:

23-NTU-CS-FL-1148

Semester:

5th- A

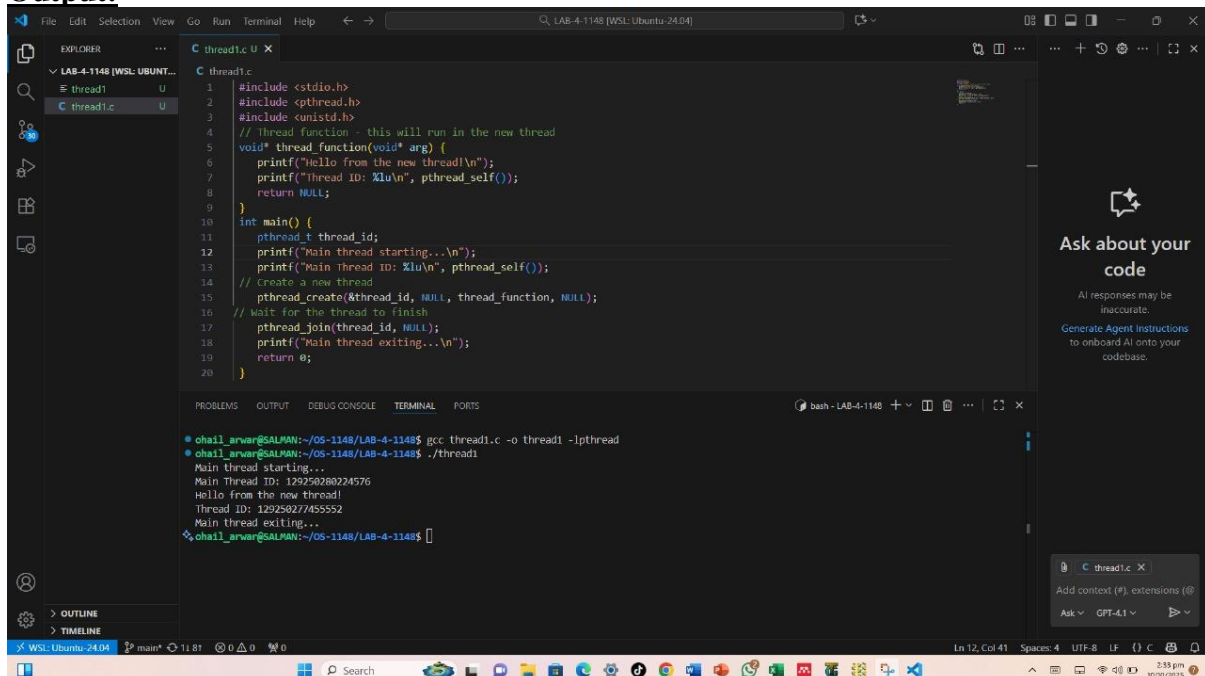
LAB-04: Introduction to Threads

Program 1: Creating a Simple Thread

Code:

```
#include <stdio.h>
#include <pthread.h>
#include <unistd.h>
// Thread function - this will run in the new thread
void* thread_function(void* arg) {
    printf("Hello from the new thread!\n");
    printf("Thread ID: %lu\n", pthread_self());
    return NULL;
}
int main() {
    pthread_t thread_id;
    printf("Main thread starting...\n");
    printf("Main Thread ID: %lu\n", pthread_self());
    // Create a new thread
    pthread_create(&thread_id, NULL, thread_function, NULL);
    // Wait for the thread to finish
    pthread_join(thread_id, NULL);
    printf("Main thread exiting...\n");
    return 0;
}
```

Output:

The screenshot shows the Visual Studio Code interface with a C file named 'thread1.c' open. The code in the editor matches the provided code block. The terminal at the bottom shows the execution of the program. It starts with the command 'gcc thread1.c -o thread1 -lpthread' and then './thread1'. The output shows the main thread starting, printing its ID (129250280224576), then the new thread prints 'Hello from the new thread!' and its ID (129250277455552), followed by the main thread printing 'Main thread exiting...'.

```
ohail_arwan@SALMAN:~/OS-1148/LAB-4-1148$ gcc thread1.c -o thread1 -lpthread
ohail_arwan@SALMAN:~/OS-1148/LAB-4-1148$ ./thread1
Main thread starting...
Main Thread ID: 129250280224576
Hello from the new thread!
Thread ID: 129250277455552
Main thread exiting...
ohail_arwan@SALMAN:~/OS-1148/LAB-4-1148$
```

Program 2: Passing Arguments to Threads

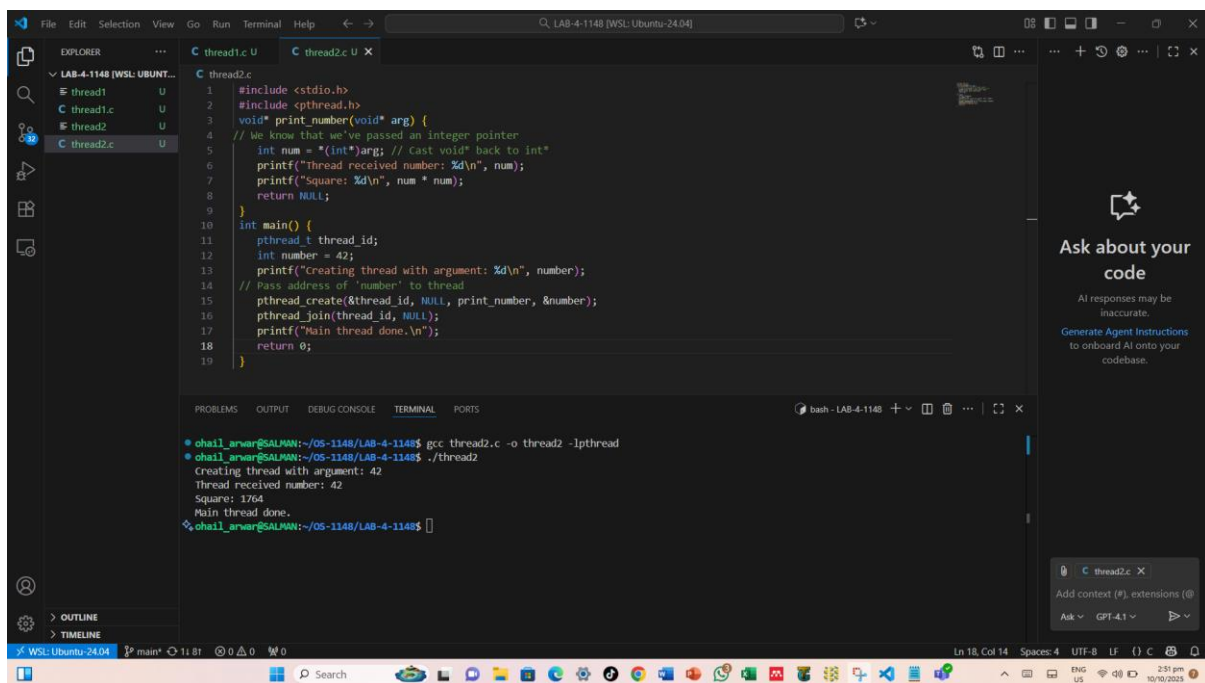
Code:

```
#include <stdio.h>
#include <pthread.h>
void* print_number(void* arg) {
```

```
// We know that we've passed an integer pointer
int num = *(int*)arg; // Cast void* back to int*
printf("Thread received number: %d\n", num);
printf("Square: %d\n", num * num);
return NULL;
}

int main() {
    pthread_t thread_id;
    int number = 42;
    printf("Creating thread with argument: %d\n", number);
    // Pass address of 'number' to thread
    pthread_create(&thread_id, NULL, print_number, &number);
    pthread_join(thread_id, NULL);
    printf("Main thread done.\n");
    return 0;
}
```

Output:



The screenshot shows a Visual Studio Code editor window with a C program in a file named `thread2.c`. The code defines a function `print_number` that takes a `void*` argument, casts it to an `int*`, prints the value and its square, and returns `NULL`. The `main` function creates a thread with the `print_number` function, passing the address of a variable `number` initialized to 42. After joining the thread, it prints "Main thread done."

The terminal output shows the compilation and execution of the program:

```
cha1l_arwar@SALMAN:~/OS-1148/LAB-4-1148$ gcc thread2.c -o thread2 -lpthread
cha1l_arwar@SALMAN:~/OS-1148/LAB-4-1148$ ./thread2
Creating thread with argument: 42
Thread received number: 42
Square: 1764
Main thread done.
cha1l_arwar@SALMAN:~/OS-1148/LAB-4-1148$
```

The right sidebar of the editor displays a chat interface with the text "Ask about your code" and a note that "AI responses may be inaccurate." Below this, there is a button to "Generate Agent Instructions" and a search bar for the chat.